AMENDMENTS TO THE CLAIMS

Claim 1 (Original): A method for building a time-sliced architecture in a spread spectrum system, comprising the steps of:

(a) analyzing a set of applications, said analyzing including the steps of:

(i) extracting real time aspects from each application in said set of applications;

(ii) determining an optimal granularity based on said real time aspects; and

(iii) adjusting said optimal granularity based on a context switching overhead; and

(b) building a specific time-sliced architecture to accommodate said range of applications based on said analyzing.

Claim 2 (Original): The method of claim 1, wherein said extracting includes the step of:

profiling fundamental processing elements in each application in said set of applications.

Claim 3 (Original): The method of claim 1, wherein said determining includes the step of:

determining a lowest level of granularity needed for each application in said set of applications.

Claim 4 (Original): The method of claim 1, wherein said adjusting includes the step of:

performing a sensitivity analysis.

Claim 5 (Original): The method of claim 4, wherein said performing includes the step of:

determining an optimal trade-off between said context switching overhead and said optimal granularity.

Claim 6 (Original): The method of claim 1, wherein said building includes the steps of:

determining a size for a data cache based on said extracting;

implementing a hierarchical caching structure in said data cache; and

applying said data cache in said specific time-sliced architecture.

Claim 7 (Original): A computer program product for building a time-sliced architecture in a spread spectrum system, comprising:

(a) logic code for analyzing a set of applications, said logic code for analyzing including:

(i) logic code for extracting real time aspects from each application in said set of applications;

(ii) logic code for determining an optimal granularity based on said real time aspects; and

(iii) logic code for adjusting said optimal granularity based on a context switching overhead; and

(b) logic code for building a specific time-sliced architecture to accommodate said range of applications based on said analyzing.

Claim 8 (Original): The computer program product of claim 7, wherein said logic code for extracting includes:

logic code for profiling fundamental processing elements in each application in said set of applications.

Claim 9 (Original): The computer program product of claim 7, wherein said logic code for determining includes:

logic code for determining a lowest level of granularity needed for each application in said set of applications.

Claim 10 (Original): The computer program product of claim 7, wherein said logic code for adjusting includes:

logic code for performing a sensitivity analysis.

Claim 11 (Original): The computer program product of claim 10, wherein said logic code for performing includes:

logic code for determining an optimal trade-off between said context switching overhead and said optimal granularity.

Claim 12 (Original): The computer program product of claim 7, wherein said logic code for building includes:

logic code for determining a size for a data cache based on said extracting;

logic code for implementing a hierarchical caching structure in said data cache; and

logic code for applying said data cache in said specific time-sliced architecture.

Claim 13 (Original): A time-sliced processor for use in a spread spectrum system comprising:

a master control unit including a time slot table and a partial sums search table;

a data cache for receiving input data; and

a plurality of finger processing elements, each element comprising:

a cache for receiving data from the data cache,

a data selector connected to an output of the cache,

a despreader connected to an output of the data selector, and

a symbol integrator connected to an output of the despreader.